



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/051,274	01/22/2002	Sunil Kunistetty	50277-2895	7586

42425 7590 03/28/2006

HICKMAN PALERMO TRUONG & BECKER/ORACLE
2055 GATEWAY PLACE
SUITE 550
SAN JOSE, CA 95110-1089

EXAMINER

PHAM, CHRYSTINE

ART UNIT	PAPER NUMBER
----------	--------------

2192

DATE MAILED: 03/28/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/051,274	Applicant(s) KUNISSETY ET AL.	
	Examiner Chrystine Pham	Art Unit 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 20 December 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1,2,5,6,9-12 and 15-22 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1,2,5,6,9-12 and 15-22 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on December 20th 2005 has been entered.
2. This action is responsive to Amendment filed on December 20th 2005. Claims 1-2, 5-6, and 11 have been amended. Claims 3-4, 7-8, 13-14 have been canceled. Claims 15-22 are new claims. Claims 1-2, 5-6, 9-12, 15-22 are pending.

Response to Arguments

3. Applicant's arguments with respect to newly added limitations "sharing markup text from a page among a plurality of users", and "analyzing the page to extract markup text, wherein the page includes at least the markup text and a set of code instructions to be executed as an application" recited in independent claims 1 and 15 have been considered but are moot in view of the new ground(s) of rejection.
4. Applicant's arguments filed December 20th 2005 with respect to Friesen's anticipation (as established in the final Office Action and the advisory action) of "loading the class

containing the pre-initialized static variable into a share, read-only memory is accessible to said plurality of users” have been fully considered but they are not persuasive.

As has been established in the final Office Action and the Advisory Action, the static variables in col.3:28-30 refer to static variables for a shopping cart discussed in the same paragraph. It is further submitted that the “shopping cart” frame (i.e., ShopFrame) is only an object or an instance of a class implementing the “shopping cart” object (see at least *class app, class ShopFrame, class ShopPanel* col.15:15-col.16:61). Thus, modifying values for static variables in a shopping cart frame or instance (e.g., user deselects an item, adds a new item, or changes quantity of a selected item) has no effect on the **class** containing the defined static variables and does not equate to modifying the **class** containing the defined static variables. In col.9:48-col.10:46, Friesen discloses a database or cache “update.txt” containing an itemPrice for its corresponding itemID. Friesen explicitly states that this cache is defined as a **static String variable** (i.e., **an array of characters**) in a class implementing the shopping cart object. In col.19:35-50, Friesen explicitly discloses **Constants.class** (which implements the shopping cart object) containing said **static String variable**. It is inherent that the **static String is pre-initialized to “update.txt” in said class**. It is further inherent and essential to Friesen’s invention that said class is loaded into a shared, read-only memory. Since if the class is not in the shared, read-only memory, itemPrice can be freely modified for any corresponding itemID. Furthermore, Friesen’s invention is directed persisting a shopping cart object from one web page to another. It is inherent that classes (containing static variables) implementing the shopping cart object are persisted in memory (i.e., **read-**

only) from one web page to another (i.e., shared by web pages), thus enables Friesen to eliminate the need to download the same classes (in order to maintain the shopping cart object) every time the user accesses a different web page. Needless to say, it is obvious in Friesen that the shared, read-only memory containing the static String variable (containing all the itemIDs and their corresponding itemPrices) is accessible to different (i.e., plurality of) users/sessions (online shoppers) as well as it being accessible between web pages for each individual user's shopping-session, each user/shopper having a unique and personalized instance of the shopping cart for each unique and personalized purchase.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.
6. Claims 1-2, 5-6, 9-12, 15-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Agrawal et al. (US 2002/0004813 A1, *Agrawal*) in view of Friesen (US 6636863, *Friesen*)

Claim 1

Agrawal teaches a computer-implemented method of sharing markup text from a page among a plurality of users in response to requests from said plurality of user (see at least *request for a document, script, blocks, data source, code* paragraph [0014]; *HTML document, HTTP request* paragraph [0015]; *612, cached blocks 614, client devices 606* FIG.6 & associated text; paragraph [0061]), said method comprising:

- Analyzing the page (see at least *Web page 202* FIG.2 & associated text; *Web page 202, script, blocks* paragraph [0032]) to extract static markup text (see at least *S51-S53* FIG.5 & associated text; *portion of Web page, static, static HTML* paragraph [0031]; *pages, different users, page blocks, current weather, common zip code, partial page caching* paragraph [0060]), wherein the page includes at least the markup text and a set of code instructions to be executed as an application (see at least *request for a document, script, blocks, data source, code* paragraph [0014]; *HTML document, HTTP request* paragraph [0015]; *Web scripting, application logic, ASP, JSP* paragraph [0030]; *combination of code and static HTML, script* paragraph [0031]);

Agrawal further teaches caching/loading said static markup text from the page into a shared, read-only memory (see at least *S53-S58* FIG.5 & associated text; *caching, selected number of blocks, web page* paragraph [0033]; paragraph [0034]; *static portion, cache memory* paragraph [0066]), wherein the shared read-only memory is accessible to said plurality of users (see at least *S59-S60, S53, S61-S62* FIG.5 & associated text; *612, cached blocks 614, client devices 606* FIG.6 & associated text; *rest of the page, shopping cart, item description, user comments, cached in memory, other users* paragraph [0061]; *shared memory cache, multiple processes, machines* paragraph [0053]; paragraph [0056]; paragraph [0064]).

Agrawal does not expressly disclose pre-initializing a static variable of a class to contain the markup text. However, *Friesen* discloses

- o pre-initializing a static variable of a class to contain the markup text (e.g., see *shopping cart, identities, prices, static variables* col.3:30-40; see *class ShopPanel, variables* col.15:10-col.16:6; see *app.class, Constants.class* col.17:37-52; see *Constants.class, Vector class, item ID, item price* col.18:40-50; see *static String variable* col.19:48-50); and
- o loading the class containing the pre-initialized static variable into a shared, read-only memory (e.g., see *programs, web pages, memory, subsequent use, shopping cart, first web page* col.2:67-col.3:15; see *update.txt, 'itemPrice', cache* col.10:11-37; see *Constants.class, Vector class, item ID, item price* col.18:40-50; see *static String variable* col.19:48-50; *itemID, itemPrice, database, cache, "update.txt, static String, JAVA class* col.9:48-col.10:45).

Agrawal and *Friesen* are analogous art because they are directed to persisting/sharing read-only information (i.e., static HTML such as itemIDs, itemPrices, or item descriptions) among a plurality of web pages within a session as well as among plurality of users (i.e., online shoppers) using JAVA programming language. It would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the teaching of *Friesen* into that of *Agrawal* for the inclusion of pre-initializing a static variable of a class to contain the markup text from the page. And the motivation for doing so would have been to eliminate the need to re-generate and download updated page each time an item is selected (i.e., placed into the shopping cart to be

purchased) (see at least *Friesen* col.1:55-col.2:5; col.2:65-col.3:27)(see at least *Agrawal* paragraph [0061]).

Claim 2

The rejection of base claim 1 is incorporated. *Friesen* further teaches storing the markup text in a resource file associated with the application (e.g., see *MICROSOFT ACCESS*, *update.txt* col.10:10-29; see *update.txt*, *Constants.class* col.19:48-50).

Claim 5

Agrawal teaches a method of initializing a first instance of an application that shares a set of markup text with other instances of the application, wherein the first instance of the application is generated by compiling code from a page that contains both the code and the set of markup text (see at least *S51-S53* FIG.5 & associated text; *portion of Web page, static, static HTML* paragraph [0031]; *pages, different users, page blocks, current weather, common zip code, partial page caching* paragraph [0060]; *request for a document, script, blocks, data source, code* paragraph [0014]; *HTML document, HTTP request* paragraph [0015]; *Web scripting, application logic, ASP, JSP* paragraph [0030]; *combination of code and static HTML, script* paragraph [0031]) in response to a request from one or more users (see at least *request for a document, script, blocks, data source, code* paragraph [0014]; *HTML document, HTTP request* paragraph [0015]; *612, cached blocks 614, client devices 606* FIG.6 & associated text; paragraph [0061]), said method comprising:

Art Unit: 2192

- executing instructions to instantiate the first instance of the application, wherein said instructions are stored on a computer-readable medium, said instructions that, when executed, cause one or more processors to perform the steps of (see at least *machine-readable medium, instructions* paragraph [0017]):
- analyzing said page to distinguish between the code and the set of markup text (see at least *combination of code and static HTML, script* paragraph [0031]; *S51-S53 FIG.5 & associated text; portion of Web page, static, static HTML* paragraph [0031]; *pages, different users, page blocks, current weather, common zip code, partial page caching* paragraph [0060]);
- accessing the markup text in the shared, read-only memory when the code from the first instance of the application is executed (see at least paragraph [0031]; FIG.5 & associated text);

Agrawal further teaches caching/loading said static markup text from the page into a shared, read-only memory (see at least *S53-S58 FIG.5 & associated text; caching, selected number of blocks, web page* paragraph [0033]; paragraph [0034]; *static portion, cache memory* paragraph [0066]), wherein the shared read-only memory is accessible to said plurality of users (see at least *S59-S60, S53, S61-S62 FIG.5 & associated text; 612, cached blocks 614, client devices 606 FIG.6 & associated text; rest of the page, shopping cart, item description, user comments, cached in memory, other users* paragraph [0061]; *shared memory cache, multiple processes, machines* paragraph [0053]; paragraph [0056]; paragraph [0064]).

Agrawal does not expressly disclose pre-initializing a static variable of a class to contain the markup text. However, *Friesen* discloses

Art Unit: 2192

- o pre-initializing a static variable of a class to contain the markup text (e.g., see *shopping cart, identities, prices, static variables* col.3:30-40; see *class ShopPanel, variables* col.15:10-col.16:6; see *app.class, Constants.class* col.17:37-52; see *Constants.class, Vector class, item ID, item price* col.18:40-50; see *static String variable* col.19:48-50); and
- o loading the class containing the pre-initialized static variable into a shared, read-only memory (e.g., see *programs, web pages, memory, subsequent use, shopping cart, first web page* col.2:67-col.3:15; see *update.txt, 'itemPrice', cache* col.10:11-37; see *Constants.class, Vector class, item ID, item price* col.18:40-50; see *static String variable* col.19:48-50; *itemID, itemPrice, database, cache, "update.txt, static String, JAVA class* col.9:48-col.10:45).

Agrawal and *Friesen* are analogous art because they are directed to persisting/sharing read-only information (i.e., static HTML such as itemIDs, itemPrices, or item descriptions) among a plurality of web pages within a session as well as among plurality of users (i.e., online shoppers) using JAVA programming language. It would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the teaching of *Friesen* into that of *Agrawal* for the inclusion of pre-initializing a static variable of a class to contain the markup text from the page. And the motivation for doing so would have been to eliminate the need to re-generate and download updated page each time an item is selected (i.e., placed into the shopping cart to be purchased) (see at least *Friesen* col.1:55-col.2:5; col.2:65-col.3:27)(see at least *Agrawal* paragraph [0061]).

Claim 6

The rejection of base claim 5 is incorporated. *Agrawal* (as modified by *Friesen*) further teaches wherein the class is not loaded into the shared, read-only memory when the other instances of the application are executed (see at least *S61-S62* FIG.5 & associated text).

Claim 9

The rejection of base claim 1 is incorporated. *Friesen* further teaches wherein the markup text includes information to be displayed to a user an annotation directing a user agent how to render the information to be displayed to the user; and the markup output by the application includes the annotation (e.g., see *shopping cart frame 12* FIG.1 & associated text; col.5:35-47; see col.6:60-col.7:10; see *applet* col.15:10-45; see *frame* col.16:19-20).

Claim 10

The rejection of base claim 1 is incorporated. *Friesen* further teaches wherein the static variable of a class is an array of characters (see at least *static String variable* col.19:48-50; *itemID*, *itemPrice*, *database*, *cache*, "*update.txt*", *static String*, *JAVA class* col.9:48-col.10:45).

Claim 11

Claim recites limitations, which have been addressed in claim 9, therefore, therefore, is rejected for the same reasons as cited in claim 9.

Claim 12

Claim recites limitations, which have been addressed in claim 10, therefore, is rejected for the same reasons as cited in claim 10.

Claims 15-22

Claims recite limitations which have been addressed in claims 2, 5, 6, 9, and 10, therefore, are rejected for the same reasons cited in claims 2, 5, 6, 9, and 10.

Conclusion

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chrystine Pham whose telephone number is 571-212-3702. The examiner can normally be reached on Mon-Fri, 8:30am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



TUAN DAM
SUPERVISORY PATENT EXAMINER